L Number	Hits	Search Text	DB	Time stamp
1	7	backup$3 same plex$2	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2003/03/12 14:10

US-PAT-NO: 6460144

DOCUMENT-IDENTIFIER: US 6460144 B1

TITLE: Resilience in a multi-computer system

DATE-ISSUED: October 1, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Ashcroft; Derek William	Altrincham	N/A	N/A	GB
Atkinson; Geoffrey	Ouston	N/A	N/A	GB
Robert	East Bierley	N/A	N/A	GB
McKirgan; Philip	Alford	N/A	N/A	GB
Tickhill; Stephen Paul				

ASSIGNEE INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
International Computers Limited	London	N/A	N/A	GB	03

APPL-NO: 09/ 385937

DATE FILED: August 30, 1999

FOREIGN-APPL-PRIORITY-DATA:

COUNTRY	APPL-NO	APPL-DATE
GB	9819523	September 8, 1998
GB	9819524	September 9, 1998
GB	9900473	January 12, 1999

INT-CL: [07] G06F011/00

US-CL-ISSUED: 714/4

US-CL-CURRENT: 714/4

FIELD-OF-SEARCH: 714/4; 714/5 ; 714/6 ; 714/10 ; 714/11 ; 714/12 ; 714/13
; 714/25 ; 714/42 ; 714/43 ; 714/47 ; 714/2 ; 714/3 ; 714/56

REF-CITED:

U.S. PATENT DOCUMENTS

PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
4371754	February 1983	De et al.	179/18EE
N/A	N/A		
4466098	August 1984	**Southard**	371/9
N/A	N/A		
5155729	October 1992	Rysko et al.	371/9.1
N/A	N/A		
5278969	January 1994	**Pashan** et al.	395/425
N/A	N/A		
5408649	April 1995	Beshears et al.	395/575
N/A	N/A		
5600808	February 1997	**Kasukawa**	395/672
N/A	N/A		
5621884	April 1997	Beshears et al.	395/182.08
N/A	N/A		
5870537	February 1999	**Kern** et al.	395/162.04
N/A	N/A		
5974114	October 1999	Blum et al.	379/9
N/A	N/A		
6167531	December 2000	**Silwinski**	714/13
N/A	N/A		
6205557	March 2001	Chong et al.	714/4
N/A	N/A		

OTHER PUBLICATIONS

Kramer, "Fault-Tolerant LANs Guard Against Malfunction, Data Loss", PC Week, vol. 4, No. 37, Sep. 15, 1987, pp. C26-30.

ART-UNIT: 2184

PRIMARY-EXAMINER: Iqbal; Nadeem

ABSTRACT:

A multi-node computer system is described which includes a number of active nodes and a standby node. Each node hosts a server installation. Each server has a system disk, and a recovery disk, which holds a synchronised recovery copy of data held on the system disk. In the event of failure of a node, a recovery process is run to reconfigure the system, by connecting the recovery disk corresponding to the failed computer to the system disk of the standby computer, and copying the contents of this recovery disk to the system disk. This causes the server in the failed node to migrate to the standby node, which

thus becomes an active node.

8 Claims, 3 Drawing figures

Exemplary Claim Number: 1

Number of Drawing Sheets: 3

BRIEF SUMMARY:

(1) BACKGROUND TO THE INVENTION

(2) This invention relates to techniques for achieving resilience in a multi-computer system.

(3) Such systems are often used to support a large number of users, and to store very large databases. For example, a typical system may consist of 8 server computers, supporting up to 50,000 users and may store one or more 300 GigaByte databases.

(4) It would be desirable to be able to provide such a system based on standard server software such as for example Microsoft Exchange running under Microsoft Windows NT. However, a problem with this is that of providing resilience to failure of one of the computers. The use of cluster technology for a system of this scale would be too expensive. Also, Microsoft Exchange is not a cluster-aware application, and it is not permissible to have two instances of Exchange on the same server (even a 2-node cluster).

(5) SUMMARY OF THE INVENTION

(6) According to the invention, there is provided a method of operating a computer system comprising a plurality of computers, a plurality of system disk units, one for each of said computers, and a plurality of further disk units, one for each of said computers, the method comprising: (a) designating a plurality of said computers as active computers and designating another of said computers as a standby computer; (b) using the further disk units to provide a synchronised recovery copy of data held on the system disk units, and (c) reconfiguring the system in the event of failure of one of the active computers, by causing the standby computer to pick up the further disk unit corresponding to the failed computer.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of a multi-node computer system embodying the invention.

FIG. 2 is a flow chart showing a recovery process for handling failure of one of the nodes of the system.

FIG. 3 is a block diagram showing an example of the system after reconfiguration by the recovery process.

(1) DESCRIPTION OF AN EMBODIMENT OF THE INVENTION

(2) One computer system in accordance with the invention will now be described by way of example with reference to the accompanying drawings.

(3) In the present specification, the following terms are used with specific meanings: Node: this means an individual computer hardware configuration. In the present embodiment of the invention, each node comprises an ICL Xtraserver computer. Each node has a unique identity number. Server: this means a specific server software installation. In the present embodiment of the invention, each server comprises a specific Microsoft NT installation. Each server has a unique server name, and is capable of being hosted (i.e. run) on any of the nodes. A server can, if necessary, be shut down and relocated to another node. System: this means a number of servers accessing a common storage unit.

(4) Referring to FIG. 1, this shows a system comprising N+1 nodes 10. In normal operation, N of the nodes are active, while the remaining one is a standby. In this example, N equals four (i.e. there are 5 nodes altogether). Each of the nodes 10 hosts a server 11.

(5) The system also includes a system administration workstation 12, which allows a (human) operator or system administrator to monitor and control the system. Each server displays its name and current operational state on the workstation 12. One or more other systems (not shown) may also be controlled and monitored from the same workstation.

(6) All of the nodes 10 are connected to a shared disk array 13. In this example, the disk array 13 is an EMC Symmetrix disk array. This consists of a large number of magnetic disk units, all of which are mirrored (duplexed) for resilience. In addition, the disk array includes a number of further disks, providing a Business Continuance Volume (BCV). A BCV is effectively a third **plex,** which can be connected to or disconnected from the primary **plexes** under control of EMC Timefinder software, running on the workstation 12. The BCV data can be synchronised with the primary **plexes** so as to provide a **backup,** or can be disconnected from the primary **plexes,** so as to provide a snapshot of the main data at a given point in time. When the BCV has been split in this way, it can be reconnected at any time and the data then copied from the primary **plexes** to the BCV, or vice versa, to resynchronise them.

(7) The system also includes an archive server 14 connected to the disk array 13 and to a number of robotic magnetic tape drives 15. In operation, the archive server periodically performs an offline archive of the data in each database, by archiving the copy of the database held in the BCV to tape. When the archive is secure, the BCV is then brought back into synchronism with the main database, before again being broken away to form the recovery BCV, using the EMC TimeFinder software.

(8) As illustrated in FIG. 1, the disk array 13 includes a number of system disks 16, one for each of the servers 11. Each system disk holds the NT operating system files and configuration files for its associated server: in other words, the system disk holds all the information that defines the "personality" of the server installation. Each of the system disks has a BCV disk 17 associated with it, holding a backup copy of the associated system disk. Normally, each BCV disk 17 is disconnected from its corresponding system disk; it is connected only if the system disk changes, so as to synchronise the two copies.

(9) In the event of failure of one of the N active nodes 10, a recovery process is initiated on the system administration workstation 12. In this example, the recovery process comprises a script, written in the scripting language associated with the Timefinder software. The process guides the system administrator through a recovery procedure, which reconfigures the system to cause the standby node to pick up the system disk BCV of the failed node, thereby relocating the server on the failed node on to the standby node and vice versa.

(10) The recovery process makes use of a predetermined set of device files, one for every possible combination of node and server. Since in this example there are five servers and five nodes (including the standby), there are 25 possible combinations, and hence 25 such device files are provided. Each of these files is identified by a name in the form n(N)_is_(S) where N is a node identity number, and S is the last three digits of the server name. (Other conventions could of course be used for naming the files). Each device file contains all the information required to install the specified server on the specified node.

(11) As illustrated in FIG. 2, the recovery process comprises the following steps:

(12) (Step 201) The recovery process first confirms the identity of the failed system with the administrator. This step is required only if more than one system is managed from the same system administration workstation.

(13) (Step 202) The recovery process then queries the administrator to obtain the identity numbers of the failed node and the standby node. The administrator can determine these node numbers using information displayed on the system administration workstation 12.

(14) (Step 203) The recovery process next queries the system administrator to obtain the name of the failed server (i.e. the server currently running on the failed node). The recovery process also automatically determines the name of

the standby server --this is a predetermined value for each system.

(15) (Step 204) The recovery process also automatically determines the device identifiers for the BCVs associated with the failed server and the standby server, using a lookup table which associates each server name with a particular device identifier.

(16) (Step 205) The recovery process then calls the BCV QUERY command in the Timefinder software, so as to determine the current states of these two BCVs. These should both be in the disconnected state.

(17) If one or both of the BCVs is not in the disconnected state, the recovery process aborts, prompting the system administrator to call the appropriate technical support service.

(18) (Step 206) If both of the BCVs are in the disconnected state, the recovery process continues by prompting the administrator to ensure that both the failed server and the standby server are shut down. The recovery process waits for confirmation that this has been done.

(19) (Step 207) When both the failed server and the standby server have been shut down, the recovery process constructs two device file names as follows:

(20) The first file name is n(W)_is_(X) where W is the node number of the standby node and X is the last three digits of the failed server's name.

(21) The second file name is n(Y)_is_(Z) where Y is the node number of the failed node and Z is the last three digits of the standby server's name.

(22) (Step 208) The recovery process then calls the Timefinder BCV RESTORE command passing it the first device file name as a parameter. This causes the BCV of the failed node to be linked to the system disk of the standby server, and initiates copying of the data from this BCV to the system disk. It can be seen that the effect of this is to relocate the server that was running on the failed node on to the standby node.

(23) The recovery process also calls the BCV RESTORE command, passing it the second device file name as a parameter. This causes the BCV of the

standby
node to be linked to the system disk of the failed server, and initiates
copying of the data from this BCV to the system disk. The effect of this is
therefore to relocate the server that was running on the standby node on to the
failed node.

(24) As an example, FIG. 3 shows the case where node 1 has failed, and where
node 4 is the standby. As shown, the BCV disk of the standby node is linked to
the system disk of the failed node, and the BCV of the failed node is linked to
the system disk of the standby

(25) While the restore commands are running, the recovery process checks for
error responses, and reports any such responses to the administrator. It also
writes all actions to a log file immediately prior to the action.

(26) (Step 209) After issuing the restore commands, the recovery process
prompts the administrator to restart the recovered server (i.e. the server
which has migrated from the failed node to the standby node), stating the new
node name it will run on. The standby node therefore now becomes an active
node.

(27) It should be noted that the restore commands run in the background and
typically take about an hour to complete. However, the recovered server can be
restarted immediately, and its data accessed, without waiting for the restore
commands to complete.

(28) (Step 210) The recovery procedure monitors for completion of the BCV
restore operations, using the Timefinder BCV Query command.

(29) (Step 211) When the restore operations are complete, the recovery
procedure issues a Timefinder BCV Split command, which disconnects the BCVs
from the system disks. Recovery is now complete, and the recovery process
terminates.

(30) Once the failed node has been fixed, it can be rebooted as required, and
will become the standby server. The recovery procedure can then be repeated if

any of the active nodes fails.

(31) Some possible modifications

(32) It will be appreciated that many modifications may be made to the system described above without departing from the scope of the present invention. For example, different numbers of disks and computers may be used. Also, the invention may be implemented in other operating systems, and using other hardware configurations. Moreover, instead of implementing the recovery procedure by means of a script, it could for example be integrated into the operating system.

CLAIMS:

What is claimed is:

1. A method of providing resilience in a multi-node computer system comprising a plurality of computer hardware nodes, including a plurality of active nodes and at least one standby node, the method comprising: (a) associating at least one system disk and at least one backup disk with each of the nodes; (b) hosting a plurality of server software installations on respective ones of the nodes, each of the server software installations being defined by information stored on a respective one of the system disks; (c) maintaining synchronized backup copies of the system disks on respective ones of the backup disks; and (d) in the event of failure of one of the active nodes, reconfiguring the system to cause the standby node to pick up the backup disk associated with the failed node, and relocating the server software installation currently hosted on the failed node onto the standby node.

2. A method according to claim 1 wherein the step of reconfiguring the system further comprises copying of information from the backup disk associated with the failed node to the system disk associated with the standby node.

3. A method according to claim 2 further including restarting the standby computer while the copying of information is being performed in the background.

4. A method according to claim 3, including the step of maintaining a set

of device files, one for each possible combination of hardware node and server
software installation, wherein the step of reconfiguring the system comprises
selecting two of the device files that correspond to the new configurations of
the failed computer and the standby computer and using the selected device
files to control reconfiguration of the system.

5. A multi-node computer system comprising: (a) a plurality of computer
hardware nodes, including a plurality of active nodes and at least one standby
node; (b) a plurality of system disks; (c) a plurality of backup disks
holding synchronized backup copies of the system disks; (d) means for
associating at least one of the system disks and at least one of the backup
disks with each of the nodes; (e) a plurality of server software installations
hosted on respective ones of the nodes, the server software installations being
defined by information stored on the system disks of the respective nodes; and
(f) means for reconfiguring the system in the event of failure of one of the
active nodes, to cause the standby node to pick up the backup disk associated
with the failed node, and relocating the server software installation currently
hosted on the failed node onto the standby node.

6. A system according to claim 5, wherein the means for reconfiguring the
system includes means for copying of information from the backup disk
associated with the failed node to the system disk associated with the standby
node.

7. A system according to claim 6 further including means for restarting the
standby computer while the copying of information is being performed in the
background.

8. A system according to claim 3, including a set of device files, one for
each possible combination of hardware node and server software installation,
wherein the means for reconfiguring the system comprises means for selecting
two of the device files that correspond to the new configurations of the failed
computer and the standby computer and for using the selected device files to
control reconfiguration of the system.

(19)日本国特許庁（ＪＰ）　(12)公開特許公報（Ａ）　(11)特許出願公開番号
特開2000－353055
（P2000－353055A）
(43)公開日　平成12年12月19日(2000.12.19)

(51)Int.Cl.'		識別記号	FI		テーマコード'(参考)
Ｇ０６Ｆ	3/06	３０４	Ｇ０６Ｆ 3/06	３０４Ｆ	５Ｂ０６５
	12/00	５３１	12/00	５３１Ｍ	５Ｂ０８２

審査請求　有　請求項の数６　ＯＬ　（全４頁）

(21)出願番号　特願平11－166346

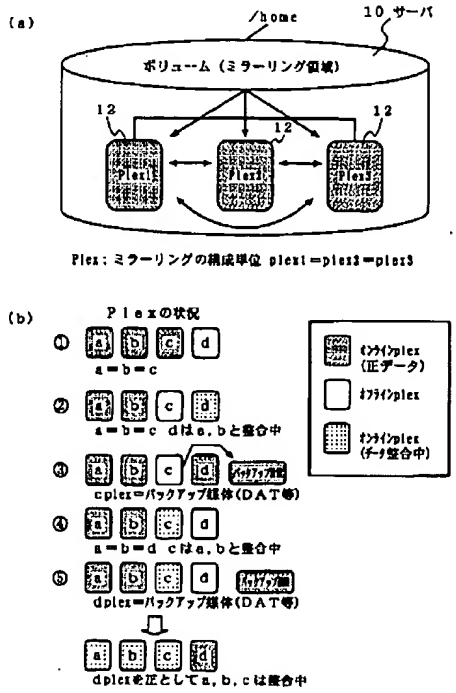
(22)出願日　平成11年６月14日(1999.6.14)

(71)出願人　390001432
日本電気ビジネスシステム株式会社
東京都港区三田３丁目14番10号
(72)発明者　大島　洋之
東京都港区三田３－14－10　日本電気ビジネスシステム株式会社内
(74)代理人　100086645
弁理士　岩佐　義幸

Ｆターム(参考)　5B065 CA11 CA13 EA02 EA12 EA23
EA31 EA33 EA34 EA40
5B082 DE05 DE06

(54)【発明の名称】　無停止システムのバックアップ方法

(57)【要約】
【課題】　サーバを停止させることなくデータを短時間でミラーリングする無停止システムのバックアップ方法を提供する。
【解決手段】　サーバ上にある無停止システムのバックアップ方法を実現するプログラムは、グループウェアのシステム領域およびデータ領域を３重化ミラーリングする。プログラムは、ｐｌｅｘ１２を切り離した後、切り離したボリューム１０を物理バックアップする。プログラムは、バックアップをおこなっている間、残りの２ｐｌｅｘで２重化ミラーリングをおこない、システムを運転させる。



1

【特許請求の範囲】
【請求項1】サーバの稼働中にミラーリングから切り離した、ミラーリングの構成単位であるｐｌｅｘからデータをバックアップすることを特徴とする無停止システムのバックアップ方法。
【請求項2】前記ミラーリングの構成を増やすことにより、運転領域およびバックアップ領域を常に前記ミラーリングすることを特徴とする請求項1記載の無停止システムのバックアップ方法。
【請求項3】グループウェアのシステム領域およびデータ領域を3重化ミラーリングし、そのうちの1つの前記ｐｌｅｘを切り離した後、切り離したミラーリング領域を物理バックアップし、バックアップをおこなっている間、前記サーバを停止することなく残りの2つの前記ｐｌｅｘで2重化ミラーリングをおこなうことを特徴とする請求項1または2記載の無停止システムのバックアップ方法。
【請求項4】物理バックアップをおこなうことで、通常バックアップよりバックアップ時間の短縮を実現することが可能であることを特徴とする請求項1から3まで記載の無停止システムのバックアップ方法。
【請求項5】ミラーリングの構成である前記ｐｌｅｘを複数持つことを特徴とする請求項1から4まで記載の無停止システムのバックアップ方法。
【請求項6】バックアップ用のｐｌｅｘを複数もちミラーリングさせ、バックアップ用ｐｌｅｘからデータを復旧させた場合、すぐにミラーリング構成で運用することが可能なことを特徴とする請求項1から5まで記載の無停止システムのバックアップ方法。
【発明の詳細な説明】
【0001】
【発明の属する技術分野】本発明は、サーバにおけるデータを、システムを停止することなくミラーリングをおこなう無停止システムのバックアップ方法に関する。
【0002】
【従来の技術】従来のバックアップ方法では、大容量システムのバックアップに時間がかかっており、システム停止を伴うバックアップでは、サービス提供時間が限られていた。つまり、24時間の無停止のサービス提供も実現できなかった。
【0003】ここで、従来の無停止型のコンピュータのデータ受付方式の一例が、特開平7-160563号公報に記載されている。この公報に記載されたオンラインバックアップ方式は、リドゥログとアンドゥログとを主記憶装置上に格納するログ蓄積手段と、データベースとログ蓄積手段とから採取されたログからデータベースの復旧をおこなうデータベース復旧手段とを備え、第1のデータベース格納手段からバックアップを採取して、第2のデータベース格納手段でトランザクション処理を続行する。これによりバックアップ処理中であっても、トランザクションを受け付けることが可能となる。

2

【0004】
【発明が解決しようとする課題】特開平7-160563号公報に記載のオンラインバックアップ方式は、第1の記憶手段に格納されているデータを第2の記憶手段へ格納することで、バックアップをおこなうものである。サーバシステムなどの場合は、格納されるデータが複数であることが多く、また1つ1つバックアップをおこなっていると、時間がかかるなどの問題が生じてしまう。
【0005】本発明の目的は、複数のＰｌｅｘ（ミラーリングの構成単位）をシステムを停止することなく短時間でバックアップする無停止システムのバックアップ方法を提供することにある。
【0006】
【課題を解決するための手段】本発明の無停止システムのバックアップ方法は、サーバの稼働中にミラーリングから切り離した、ミラーリングの構成単位であるｐｌｅｘからデータをバックアップすることを特徴とする。
【0007】また、前記ミラーリングの構成を増やすことにより、運転領域およびバックアップ領域を常に前記ミラーリングすることを特徴とする。
【0008】，
【発明の実施の形態】本発明は、24時間無停止システムのサーバ運転中に、ボリュームの物理バックアップをおこなうことができるものである。この物理バックアップは、2重化ミラーリングを実現するものである。ボリュームのデータは、ミラーリングされることにより、不慮のＤＩＳＫ障害にもデータを失うことがないように対応できる。また、物理バックアップをおこなうことは、通常バックアップよりバックアップ時間の短縮を実現することが可能である。本発明は、無停止でシステム稼働中にバックアップができないシステムが対象となる。
【0009】本発明の実施例を図1を参照し説明する。図1は、本発明の実施例の構成を示す図である。サーバ上にある無停止システムのバックアップ方法を実現するプログラムは、グループウェアのシステム領域およびデータ領域を3重化ミラーリングし、そのうちの一つのｐｌｅｘ12を切り離した後、切り離したボリューム10を物理バックアップする。プログラムは、バックアップをおこなっている間、残りの2ｐｌｅｘで2重化ミラーリングをおこない、システムを運転させる。この方法により、グループウェアのサーバは、運転しながらバックアップをおこなうことができる。ここでｐｌｅｘ12は、ミラーリングの構成単位のことである。このプログラムは、複数のｐｌｅｘでミラーリングを構成する（2ｐｌｅｘであれば、2重化ミラーリング、3ｐｌｅｘであれば、3重化ミラーリングとなる）。
【0010】次に、本発明の実施例の動作を説明する。サーバ上に存在する無停止システムのバックアップ方法を実現するプログラムは、グループウェアのシステム運

転（３重化ミラーリング）を開始する。プログラムは、１ｐｌｅｘをミラーリングシステムから切り離す。プログラムは、グループウェアのサーバを停止する。プログラムは、ｓｙｎｃコマンドにてメモリキャッシュ情報をＤＩＳＫへ書込みｐｌｅｘを切り離し、別ボリュームへ組み込む。またこのボリュームは、バックアップボリュームと呼ぶ。

【００１１】次に、プログラムは、グループウェアのシステムを起動する。プログラムは、グループウェアのサーバを起動する。このとき、グループウェアのシステムは、２重化ミラーリングとなる。プログラムは、バックアップを開始する。プログラムは、バックアップボリュームを物理バックアップする。このとき、グループウェアのシステムは２重化ミラーリングで稼働中である。プログラムは、バックアップが終了となり、バックアップボリュームのｐｌｅｘをもとのボリュームに戻す（オンライン中で可能である。）。このとき、グループウェアのシステムの停止はいらない。

【００１２】サーバは、３重化ミラーリングシステムへ元に戻したｐｌｅｘへデータ整合を処理おこなう。このとき、グループウェアのシステムは、２重化ミラーリングである。ここでボリュームとは、ＯＳがファイルシステムとして認識できる単位のことである。

【００１３】次に、本発明の他の実施例を説明する。図１（ｂ）に示すようにミラーリング構成をｎ個化する。ここではＰｌｅｘが4つとする。サーバは、バックアップ用のｐｌｅｘを常にオフライン化しておくことによりデータの復旧時間を大幅に短縮できる。

【００１４】ａ，ｂ，ｃ，ｄというｐｌｅｘが4つある場合、はじめプログラムは、ａ，ｂ，ｃでミラーリングをおこなっている（図１（ｂ）の①）。プログラムは、バックアップをおこなうタイミングで、ｃｐｌｅｘを切り離すが、そのタイミングで、ｄｐｌｅｘを組み込み対象システム領域は常に３重化ミラーリングとしておく（図１（ｂ）の②）。ｃｐｌｅｘは、ＤＡＴ等の媒体にバックアップをとったあと、そのままオフラインにしておく（図１（ｂ）の③）。この状況では、あるタイミングのバックアップデータは、媒体とＤＩＳＫ（ｃｐｌｅｘ）に存在することになる。

【００１５】次のバックアップのタイミングで、プログラムはｄｐｌｅｘを切り離し、ｃｐｌｅｘを組み込む（図１（ｂ）の④）。この方法を繰り返すことにより、常に媒体とＤＩＳＫ上（ｃｐｌｅｘかｄｐｌｅｘ）にあるバックアップデータが存在することになる。データを復旧する場合は、既存のデータ領域（３重化ミラーリング領域）のｐｌｅｘを全て切り離し、バックアップデータのあるｐｌｅｘ（ｃかｄ）を組み込むことにより、対象システム環境はバックアップ時点の状態となる（図１（ｂ）の⑤）。ＤＡＴ等の媒体からデータを戻す場合は、数時間の要するが、この方法の場合、数分で作業が完了する。

【００１６】また、この方法の応用例として、バックアップ用のｐｌｅｘを複数もちミラーリングさせることも考えられる。この場合、バックアップｐｌｅｘからデータを復旧させた場合、すぐにミラーリング構成で運用することが可能である。

【００１７】

【発明の効果】本発明の無停止システムのバックアップ方法は、通常、バックアップをおこなう場合にシステムを停止させるシステムにおいて、システム稼働中でもバックアップが可能となる。また、ＤＩＳＫをミラーリングすることにより、ＤＩＳＫ交換時は運転を停止する必要あるが、不慮のＤＩＳＫ障害が発生しても、運転を続行可能であり、データの復旧作業を省くことができる。

【図面の簡単な説明】

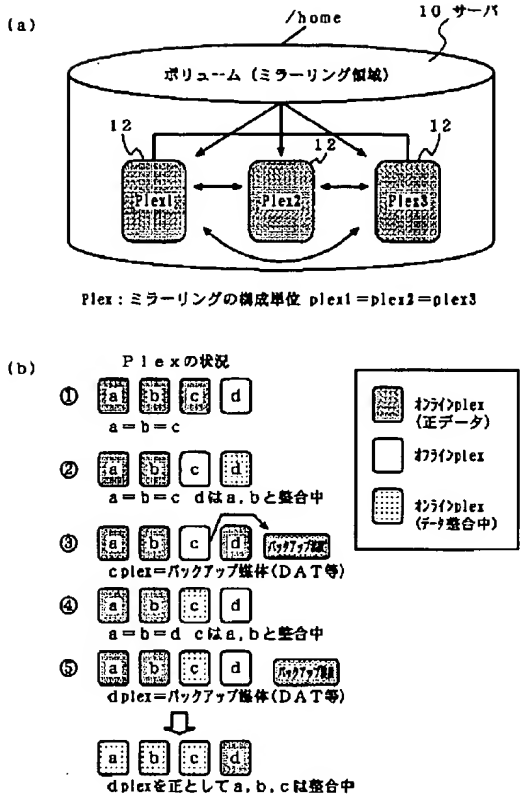
【図１】本発明の実施例の構成を示す図である。

【符号の説明】

１０　ボリューム

１２　Ｐｌｅｘ

【図1】



【手続補正書】
【提出日】平成１１年１１月１１日（１９９９．１１．１１）
【手続補正１】
【補正対象書類名】明細書
【補正対象項目名】特許請求の範囲
【補正方法】変更
【補正内容】
【特許請求の範囲】
【請求項１】サーバの稼働中にミラーリングから切り離した、ミラーリングの構成単位であるｐｌｅｘからデータをバックアップすることを特徴とする無停止システムのバックアップ方法。
【請求項２】前記ミラーリングの構成を増やすことにより、運転領域およびバックアップ領域を常に前記ミラーリングすることを特徴とする請求項１記載の無停止システムのバックアップ方法。
【請求項３】グループウェアのシステム領域およびデータ領域を３重化ミラーリングし、そのうちの１つの前記ｐｌｅｘを切り離した後、切り離したミラーリング領域を物理バックアップし、バックアップをおこなっている間、前記サーバを停止することなく残りの２つの前記ｐｌｅｘで２重化ミラーリングをおこなうことを特徴とする請求項１または２記載の無停止システムのバックアップ方法。
【請求項４】物理バックアップをおこなうことで、通常バックアップよりバックアップ時間の短縮を実現することが可能であることを特徴とする請求項１，２または３記載の無停止システムのバックアップ方法。
【請求項５】ミラーリングの構成である前記ｐｌｅｘを複数持つことを特徴とする請求項１，２，３または４記載の無停止システムのバックアップ方法。
【請求項６】バックアップ用のｐｌｅｘを複数もちミラーリングさせ、バックアップ用ｐｌｅｘからデータを復旧させた場合、すぐにミラーリング構成で運用することが可能なことを特徴とする請求項１，２，３，４または５記載の無停止システムのバックアップ方法。

CLIPPEDIMAGE= JP02000353055A

PAT-NO: JP02000353055A

DOCUMENT-IDENTIFIER: JP 2000353055 A

TITLE: BACKUP METHOD FOR NO-INTERRUPTION SYSTEM

PUBN-DATE: December 19, 2000

INVENTOR-INFORMATION:

NAME	COUNTRY
OSHIMA, HIROYUKI	N/A

ASSIGNEE-INFORMATION:

NAME	COUNTRY
NIPPON DENKI BUSINESS SYST KK	N/A

APPL-NO: JP11166346

APPL-DATE: June 14, 1999

INT-CL (IPC): G06F003/06;G06F012/00

ABSTRACT:

PROBLEM TO BE SOLVED: To provide a backup method for a no-interruption system
for mirroring data for a short time without interrupting a server.

SOLUTION: A program which realizes a **backup** method of a non-interruption on a
server triple mirrors a system area and a data area of group ware. The program
physically backs up a separated volume 10 after a **plex** 12 is cut out. The
program performs double mirroring by remaining 2 **plex while backup** is being
performed and has the system drive.